IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.	: 10/524,554	Confirmation No.	: 9119
First Named Inventor	: Richard EVANS		
Filed	: September 7, 2005		
TC/A.U.	: 2624		
Examiner	: Katrina R. Fujita		
Docket No.	: 038819.55861US		
Title	: Video Motion Anomaly Detector		

DECLARATION OF RICHARD JOHN EVANS UNDER 37 C.F.R. § 1.131

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

I, Richard John Evans, hereby declare that:

1. I am the inventor of the above-identified application.

2. I have reviewed the claims currently pending in the above-identified application, including the claim amendments that are being submitted along with this Declaration.

3. Based on information and belief the present application claims priority to two British patent applications filed on August 15, 2002 and November 18, 2002.

4. I have reviewed the article "Semantic Interpretation of Object Activities in a Surveillance System" by Lou et al. ("the Lou article"), which is attached as Exhibit A. The Lou article itself only indicates a copyright date of 2002.

5. Based on information and belief the Lou article was published on December 10, 2002, which is after the filing dates of the two British patent applications from which the present application claims priority. (See Exhibit B).

6. Based on information and belief the Lou article was published as part of the 16th International Conference on Pattern Recognition, which was held on August 11-15, 2002. (See Exhibit C). I have no knowledge as to whether the Lou article, or any of the discussions in the article, were presented or were otherwise publicly available at this conference.

7. The invention disclosed and claimed in the above-identified application, including at least claims 1 and 15 thereof, was conceived prior to the August 11, 2002.

8. Conception of the invention is evidenced by the attached copy of an invention disclosure that I prepared prior to August 11, 2002. (Exhibit D).

9. I have reviewed the Declaration by Clive French, the British patent attorney who prepared the British patent applications at my direction.

10. Based on information and belief, including at least the Declaration by Mr. French, from just prior to August 11, 2002 until August 15, 2002, the British patent application was being prepared for filing with the British Patent Office, was transmitted to the British Patent Office on August 14, 2002, and was accorded a filing date of August 15, 2002.

11. All statements made herein of my own knowledge are true and that all statements made on in formation and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine and/or imprisonment, or both under Section 1001, Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issuing thereon.

2 April 2009
Date

Richard Evans
Richard John Evans
RICHARD JOHN EVANS

EXHIBIT A

Semantic Interpretation of Object Activities in a Surveillance System*

Jiangung Lou, Qifeng Liu, Tieniu Tan and Weiming Hu
National Lab of Pattern Recogniton, Institute of Automation, CAS, Beijing, China
Email: *{jglou qfliu}@nlpr.ia.ac.cn*

Abstract

Activity analysis and semantic interpretation of tracked targets in a dynamic image sequence has recently attracted more attentions in computer vision. In this paper, a framework for semantic interpretation of vehicle and pedestrian's behaviors is proposed for practical applications in visual traffic surveillance. The trajectories recorded in the visual tracking process are analyzed using dynamic clustering and classification on which high level semantic interpretation is based. Experimental results are presented to illustrate the performance of the proposed algorithm.

1. Introduction

Traditional computer vision research usually focuses on problems such as feature extraction, image segmentation, object recognition, visual tracking, and so on. However, the most important and challenging task of computer vision is to understand and semantically interpret the contents in still images or dynamic image sequences just like humans do. An advanced visual surveillance system should be able to interpret what is happening in the dynamic scene, raise warning if some abnormal events occur, and also predict future actions of the tracked targets. In recent years, semantic interpretation of image and video has become an active topic in computer vision[1].

The main problem in images or videos' semantic interpretation is to construct a mapping [1] from images or videos into the human's conceptual space. The domain of a conceptual interpretation may be a still image or a dynamic image sequence, and the result of the interpretation may have many forms, in details or in abstraction, and can be described in natural language or symbolic models.

1.1 Related work

In recent years, many researchers have studied this problem. Some significant projects are presented in special issues [1, 10]. Bayesian network [6], neural network (NN) and Hidden Markov Model (HMM) are very popular methods in temporal sequence analysis, event and behaviour recognition. Remagnino et al. [2] propose a visual event interpreting system to describe the behavior of pedestrians and vehicles in a traffic scene, and the system is based on agent-orientated Bayesian network which can give annotations for the events in natural language. Each object agent is created to handle the event raised by one target, and an interaction agent is created when the distance between two targets is below a threshold. Sumpter et al. [3] use a three-layer neural network with a feedback mechanism to classify and predict the behaviors of tracked targets. The first neural network can automatically classify the trajectories of moving targets in the spatial feature space, then the resulting information is delivered to the second layer which is a leaky layer and also imports feedback signals. The main task of the third layer is to do prediction and to generate the final activity patterns. Fernyhough et al. [4] propose an automatically learning algorithm using a qualitative spatio-temporal model. Dance et al [5] have realized an image interpretation system named "SOO-PIN" based on a belief network which can interpret vehicle's behaviours in traffic intersections. Stauffer et al [11] use a co-occurrence statistics following a vector quantization operation to create a hierarchical binary-tree classification.

Trajectories are often used in semantic interpretation for dynamic image sequences [3, 7, 8]. Each trajectory records not only the position sequence of the tracked target, but also the speed, acceleration and direction of the target at each position. Such trajectory has enough information to be used in activity analysis. Following this, our work is also based on trajectories. In this paper, a framework for semantic interpretation of vehicles and pedestrians' behaviors is proposed for applications in visual traffic surveillance. The trajectories recorded in a visual tracking module are analyzed using a classification tree.

The remainder of this paper is arranged as follows. In Section 2, we outline the low level visual tracking system. In Section 3, we propose an activity pattern learning algorithm which has a tree structure and can be automatically constructed. By on-line classification, we can predict the future action of the tracked target and give alert when an abnormal event has happened. A simple rule based semantic description generation method is described in Section 4 which can generate the interpretations of the tracked targets' behaviours in natural language. Finally, we draw some conclusions and discuss further work.

2. Trajectory acquisition

In a visual surveillance system, the trajectories of

* This work is supported in part by NSFC (Grant No. 69825105).

1051-4651/02 $17.00 © 2002 IEEE

moving targets are acquired using low level tracking algorithms. In [9], we have described a vehicle tracking system with a static and pre-calibrated camera which can be easily extended to have ability for tracking people because people and vehicles usually have different size and aspect ratio. In this paper, we assume that the trajectories of tracked targets have been obtained, and the following sections only focus on semantic interpretation. It should be mentioned that in this work, the trajectories are generated by recording the position, speed and direction of the target at each frame. We do not record accelerations of targets because that noise in targets' positions can be amplified in acceleration information, which makes the acceleration information very unreliable.

3. Learning of trajectory patterns

3.1 Similarity between trajectories

Trajectory pattern analysis which can automatically classify the trajectories into several patterns is an important way for activity interpretation. As mentioned above, we can often analyze the activities of the tracked target by analyzing the target's route, speed and other dynamic information contained in the target's trajectory. In our system, we designed a classification tree as illustrated in Fig. 1 with three layers. We use spatial information to cluster trajectories into clusters and then use dynamic information to classify the trajectories in every cluster into classes. Identical clustering algorithms are used in these operations.

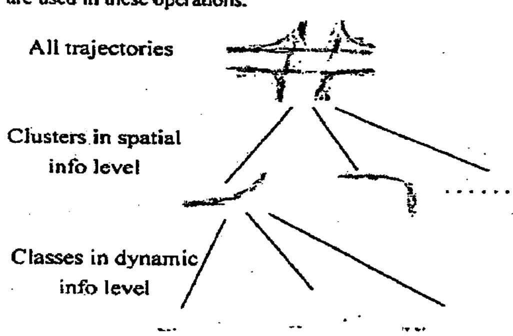


Figure 1. Classification tree

How to measure similarity between two trajectories is the first problem we should tackle before we can analyze the trajectory's spatial and dynamic information. In [3, 8], the authors use vector quantization to realize feature mapping in the Euclidean space which is expanded by the position and speed of the target. The algorithm does not use the global information of trajectories, it just does feature mapping in an unlimited feature space. In [4], the authors utilize the percentage of overlapped pixels to measure similarity between trajectories, and an 80% overlap is assumed to identify the same trajectory class. It can be regarded as a global trajectory classification, but a simple threshold will sometime fail because of noise in the trajectory which is provided by a tracking system. After all, there are some open problems in visual tracking such as occlusion.

In our work, we define a distance formulation to measure the spatial similarity between trajectories which is similar to Hausdorff distance. This distance can be considered as global information of trajectories. Given two trajectories A and B, where A has t points and B has T points, their spatial distance D_c can be defined as:

$$D_c = \min_{A,B} \{D_{A,B}, D_{B,A}\} \tag{1}$$

$$D_{A,B} = \max_{i=0..t}\{\min_{j=0..T}(d_{i,j})\}$$

$$D_{B,A} = \max_{i=0..T}\{\min_{j=0..t}(d_{i,j})\}$$

where $d_{i,j}$ is the Euclidean distance from the position of point i in one trajectory to point j in the other trajectory.

We also define a metric described below to measure similarity of trajectories' dynamic information.

$$D_t = \min_{A,B} \{DV_{A,B}, DV_{B,A}\} \tag{2}$$

where

$$DV_{A,B} = \frac{\sum_{\text{all } i \text{ in } A} \mathrm{dv}_{i,j}}{t+1}, \quad j = \arg\min_{0 \le j \le T} (d_{i,j})$$

$$DV_{B,A} = \frac{\sum_{\text{all } i \text{ in } B} \mathrm{dv}_{i,j}}{T+1}, \quad j = \arg\min_{0 \le j \le t} (d_{i,j})$$

and $dv_{i,j}$ is the difference from the speed of point i in one trajectory to point j in the other trajectory.

3.2 Off-line trajectory pattern generation

In this section, the clustering method used in our classification tree is discussed. With the definition of similarities described above, a C-Mean like clustering method is adopted which has been adapted to work with our problem.

1. To initialize centers. For clustering in the spatial information level, we simply select a threshold ρ (about 4 meters). Then, a subset of the trajectories x_k is chosen as the initial centers in such a way that every two trajectories x_i and x_j in the subset satisfy $D_c(x_i, x_j) \ge \rho$. For clustering in dynamic information level, the distribution density of every trajectory sample is evaluated by counting the number of trajectories around it, while the trajectories which have heavy densities are selected to form the initial centers.

2. All trajectory samples are classified to the class whose center is the nearest one in all classes.

$$k_t(x_t) = \arg\min_k d(x_t, x_k) \tag{3}$$

Label all elements in cluster k as $C_k = \{x_t : k_t(x_t) = k\}$, where xk is the center of cluster k.

3. For each cluster k, find a new representative (just like a center) which is the element in that cluster that has the minimal distance to all other elements in that cluster:

$$x_k \leftarrow \arg\min_{x \in C_k} \max_{x' \in C_k - \{x\}} d(x, x') \tag{4}$$

4. Repeat Step 2 and 3 until there is no change between two consecutive iterations.

5. Set a weight for every cluster, $\omega_k = \frac{N_k}{N}$ where N_k is the number of elements in cluster k, and N is the total number of samples. These weights manifest the frequency of clusters. Because the samples are selected randomly, they can be approximately regarded as the prior probabilities of clusters. We will use them in the following subsections.

3.3 Action analysis

In practical systems, a target often conducts several different actions in different segments of one trajectory. To analyze such actions, we introduce a trajectory segment analysis method to every trajectory class based on HMM similar to [7]. Each trajectory is divided into several small segments, and each small segment has 20 points. We also assign the action of the tracked target in each segment to four basic types: *Move Forward, Turn Right, Turn Left* and *Stop*. There are two main differences between our method and [7]. The first one is that we segment every trajectory into fractions which contain some small segments and then model each fraction by HMM (because it is observed that the activity pattern has significant difference at different fractions of the same trajectory). The second difference is, in our work, the curvatures of a trajectory are obtained by comparing translational speeds and angular speeds. In the low level stage, we have recorded translational speeds and angular speeds of targets in their trajectories. The curvature of a small segment can be easily obtained by $\kappa = \omega / \nu$, where ω is the mean of angular speed in that segment and υ is the mean of translation speed. For any segment in a trajectory, we can obtain a curvature value. Thus the curvature values of all segments can make up a curvature sequence, which can be used to segment the trajectory into several fractions in the curvature space by a threshold 0.1 (we assume that the target is turning its direction when the curvature value is larger than 0.1). A very simple thresholding operation described below is adopted in our system (when the speed is very slow, κ will be not stable, and we simply treat this situation as "Stop").

Move Forward	$-0.1<\kappa<0.1$ and $\upsilon>0.5$
Turn Right	$\kappa>0.1$ and $\upsilon>0.5$
Turn Left	$\kappa<-0.1$ and $\upsilon>0.5$
Stop	$\upsilon<0.5$

Through this step, all trajectories in each trajectory class are segmented into fractions which are labelled as "Move Forward", "Turn Right" or "Turn Left".

For example, in Fig. 2, there are some trajectory classes which are clustered by our algorithm. In the figure, the green segments represent "Turn Left Fraction", the blue and red segments represent "Forward Fraction" and "Turn Right Fraction" respectively. The image sequence used in our experiments is captured from a high building top with a Panasonic® video camera, and more than 400 trajectories are used in learning process (the total number of classes is 17, but we only list some of them because of the limitation of this paper's space).

Figure 2. Trajectory Classes

3.4 On-Line classification and adaption

During off-line learning, a classification tree is obtained. Now we do on-line classification with it. On-line classification here means to classify the activity pattern of a target when the target is being tracked and the trajectory is incomplete. For every new input trajectory, Bayesian classifier is implemented to do the classification. Here, we denote the distance from one point in a trajectory A to the corresponding point in the representative trajectory as $d_i = \min_{1 \le j \le T}(d_{i,j})$. In addition, we assume that these d_i, i=1...t satisfy Gaussian distribution. This assumption is reasonable when the two trajectories belong to the same class. Furthermore, the joint probability density is

$$p(x|k) = \prod_{i=1}^{t} p(x_i|k) \tag{5}$$

where $p_i(x|k)$ is the Gaussian distribution density of point x in a trajectory which belongs to class k. The parameters of these Gaussian distributions can be estimated by calculating the scatter matrix in each cluster. The post probability density will be:

$$p(k|x) = \frac{P(k) \cdot p(x \mid k)}{\sum_{\forall k} P(k) \cdot p(x \mid k)} \tag{6}$$

where $P(k)$ is the prior probability of class k, and can be substituted by ω_k mentioned above. The trajectory will be classified into the class which gives significant larger post-probability than all other classes. The next action of the tracked target can be predicted because it is assumed that the target will follow the similar behavior

to the class which it belongs to.

Once the new trajectory is completed and classified, the classification tree should be updated. We update the weight, Gaussian distribution parameters and the representative of the class which the new trajectory belongs to. Sometimes, no existing class can give significant post-probability for the tracked target's trajectory. If so, we can assert that this target is conducting an abnormal behavior, and a warning can be raised. At the same time, a new class leaf is added into the tree at a proper layer. Thus, the classification tree will vary over time. Sometimes two neighbouring classes will converge into one class. We should examine all neighbouring classes after every hour's running to determine whether to merge them.

4. Generating natural language description

Even though the activity patterns have been obtained, to generate natural language descriptions we must import some grammar rules and establish the mapping from activity patterns to words in natural language.

We introduce a simple grammar to generate natural language descriptions. Because in most surveillance scenarios, the system is often asked questions like "Who does what at where? And How?" To design a system which can answer such questions needs only a simple grammar rule. The rule is:

(The Obj) (Action) in (The place name) [at (high/low/middle) speed].

The contents in square brackets are optional and the contents in parenthesis should be substituted with the information provided by the above modules. For example, "Vehicle 1 is parked in the parking lot."

We integrate the map of the real scene with the activity map to fill the place name and also to establish mapping from activity to language (also called *Verb selection*). A typical rule like this, *if target stops in the parking lot, then output action as "is parked"*.

The system does not output the semantic description at every frame, and the output module is only activated when one of the following conditions is satisfied:

1. A new action is happening

2. The target is entering a new region

3. An abnormal event is happening

For example, in Figure 3, we demonstrate our algorithm in a real world scene. When a car enters the view and then is parked in the parking lot, the system gives the natural language description.

5. Conclusion and Further work

In this paper, we have proposed an approach which can automatically learn the activity patterns and give semantic interpretations for the tracked targets. A tree like structure is implemented to transform image data into conceptual and linguistic forms based on activity pattern analysis. The aim of this work is simultaneous semantic interpretation.

The work presented in this paper is being extended in many ways. The grammar rule database should be enriched, and a mechanism to handle interactions between objects is under consideration.

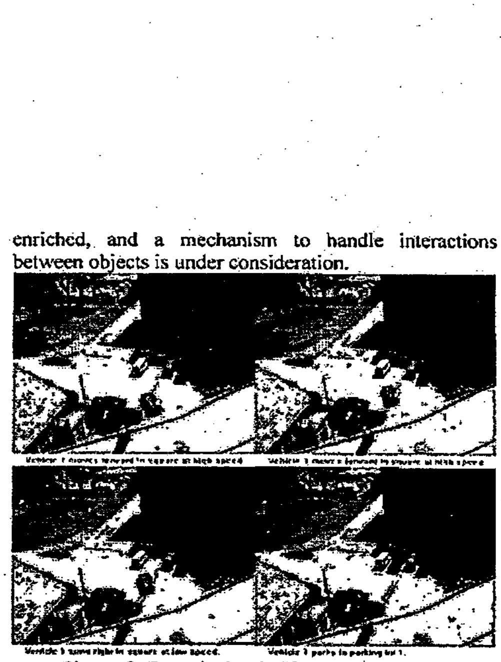


Figure 3. Description in Natural Language

Reference

[1] H. Buxton, A. Mukerjee, Conceptualizing Images, Image and Vision Computing, vol.18, 79, 2000.

[2] P.Remagnino, T.Tan and K.Baker, Agent Orientated Annotation in Model Based Visual Surveillance, pp857-862, ICCV 98.

[3] N. Sumpter, A. Bulpitt, Learning Spato-temporal Patterns for Predicting Object Behaviour, Image and Vision Computing, vol.18, 697-704, 2000.

[4] J.Fernyhough, A.G. Cohn, D.C.Hogg, Constructing Qualitative Event Model Automatically from Video Input, Image and Vision Computing, vol.18, 81-103, 2000.

[5] S Dance and T Caelli, On the Symbolic Interpretation of Traffic Scenes, Proc. of the ACCV93, pp 798-801, Osaka, 1993.

[6] T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russel, and J. Weber. Automatic Symbolic Traffic Scene Analysis using Belief Networks. pages 966--972. Proc. 12th National Conf. in AI, 1994.

[7] R. Fraile and S. J. Maybank. Vehicle Trajectory Approximation and Classification. Proc. Of British Machine Vision Conf., 1998.

[8] N. Johnson and D. C. Hogg. Learning the Distribution of Object Trajectories for Event Recognition. Image and Vision Computing. 14:609--615, 1996.

[9] J. Lou, H. Yang, W. Hu, and T. Tan, Visual Vehicle Tracking using An Improved EKF, Proc. of ACCV2002, vol.2, pp. 296-301, Jan. 2002.

[10] Robert T. Collins, Alan J. Lipton, and Takeo Kanade, Introduction to the Special Section on Video Surveillance, IEEE Trans. on PAMI, Vol. 22, NO. 8, Aug. 2000.

[11] Chris Stauffer, Eric Grimson, Learning Patterns of Activity Using Real-Time Tracking, IEEE Trans. on PAMI, vol.22, no.8, 747-757, 2000.

EXHIBIT B

Semantic interpretation of object activities in a surveillance system

Jiangung Lou Qifeng Liu Tieniu Tan Weiming Hu
Inst. of Autom., Acad. Sinica, Beijing, China;

This paper appears in: Pattern Recognition, 2002. Proceedings. 16th International Conference on
Publication Date: 2002
Volume: 3, On page(s): 777- 780 vol.3
ISSN: 1051-4651
ISBN: 0-7695-1695-X
INSPEC Accession Number: 7462226
Digital Object Identifier: 10.1109/ICPR.2002.1048115
Current Version Published: 2002-12-10

Abstract
Activity analysis and semantic interpretation of tracked targets in a dynamic image sequence has recently attracted more attentions in computer vision. In this paper, a framework for semantic interpretation of vehicle and pedestrian behaviors is proposed for practical applications in visual traffic surveillance. The trajectories recorded in the visual tracking process are analyzed using dynamic clustering and classification on which high level semantic interpretation is based Experimental results are presented to illustrate the performance of the proposed algorithm.

Index Terms
Available to subscribers and IEEE members.

References
Available to subscribers and IEEE members.

Citing Documents
Available to subscribers and IEEE members.

EXHIBIT C

ICPR archive

Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02) Volume 3 - Volume 3

2002 *August 11 - 15, 2002*

Additional Information: full citation

Table of Contents

EXHIBIT D

Video Motion Anomaly Detector (VMAD) – Description

Richard Evans, [redacted]

Technical Problem Addressed

The Video Motion Anomaly Detector addresses the problem of automatically detecting events of interest to operators of CCTV systems used in security, transport and other applications, processing CCTV images. The detector may be used in an number of ways, for example to raise an alarm, summoning a human operator to view video data, or to trigger selective recording of video data or to insert an index mark in recordings of video data.

Background to the Problem

Closed circuit television (CCTV) is widely used for security, transport and other purposes. Examples applications include the observation of crime or vandalism in public open spaces or buildings (such as hospitals and school), intrusion into prohibited areas, monitoring the free flow of road traffic, detection of traffic incidents and queues, detection of vehicles travelling the wrong way on one-way roads.

The monitoring of CCTV displays (by human operators) is a very laborious task however and there is considerable risk that events of interest may go unnoticed. This is especially true when operators are required to monitor a number of CCTV camera outputs simultaneously. As a result in many CCTV installations, video data is recorded and only inspected in detail if an event is known to have taken place. Even in these cases, the volume of recorded data may be voluminous and the manual inspection of the data may be laborious. Consequently there is a requirement for automatic devices which process the video images and raise an alarm signal when there is an event of interest. The alarm signal can be used either to draw the event to the immediate attention of an operator, to place an index mark in recorded video or to trigger selective recording of CCTV data.

Some automatic event detectors have been developed for CCTV systems, though few of these are very successful. The most common devices are called video motion detectors (VMDs) or activity detectors, though they are generally based on simple algorithms concerning the detection of changes in the brightness of the video image - not the actual movement of imaged objects. For the purposes of detecting changes in brightness, the video image is generally divided into a grid of typically 16 blocks horizontally and vertically (i.e. 256 blocks in total). There several disadvantages of these algorithms: 1) they are prone to false alarms, for example when there are changes to the overall levels of illumination, 2) they are unable to detect the movement of small objects, because of the block-based processing, 3) they cannot be applied if the scene normally contains movement objects which are not of interest. These disadvantages can be reduced to a limited extent by additional processing logic, but the effectiveness of standard VMDs is inherently limited by the use of change detection as the initial image-processing stage.

There is another type of detection device, which is characterised by the use of complex algorithms involving image segmentation, object recognition and tracking and alarm decision rules. Though these devices can be very effective, they are generally expensive systems designed for use in specific applications and do not

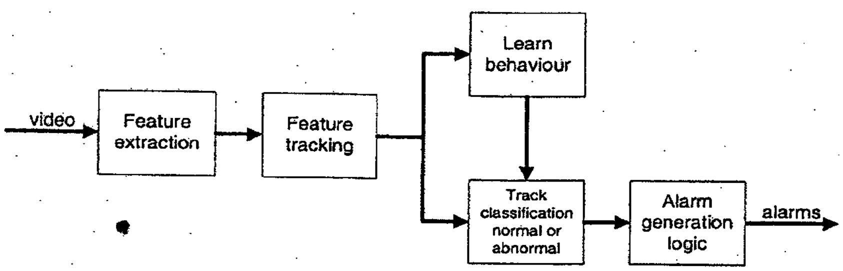
perform well without careful tuning and setting-up, and may not work at all outside of a limited range of applications for which they were originally developed.

As far as is known, the closest thing to the present invention is (in some respects) a device patented by inventors Wade & Jeffrey (Patent No: US6081606, "Apparatus and a method for detecting motion within an image sequence"). It is not known if their invention has been used in any commercial product. Briefly in their invention, motion within the image is calculated by correlating areas of one image with areas of the next image in the video to generate a flow field. The flow field is then analysed and an alarm raised dependent on the observed magnitude and direction of flow. This invention differs significantly from the video motion anomaly detector in that it is not feature based, and alarms are not generated on the basis of abnormal behaviour.

Solution to the problem

The video motion anomaly detector extracts and tracks point-like features in video images and raises an alarm when a feature (or features) is (or are) behaving abnormally compared with the behaviour of features observed over a period of time. By "behaviour" we mean the movement of features in different parts of the video image. For example, rapid movement of features in a particular direction in one part of the field of view may be normal, but it may be abnormal if it occurred in another part of the field of view where the normal behaviour is slow movement. Similarly, rapid movement in the same part of the field of view may be abnormal if the movement is in a different direction.

The following diagram shows the main processing stages in the video motion anomaly detector.



The *feature extraction* stage locates point-like features in each processed image in the video image sequence. A suitable feature has been developed by Harris (Patent No: GB2218507, "Digital Data Processing")

The *feature tracking* stage tracks features so that each point-like feature can be described by its current point and its estimated velocity in the image.

The *learn behaviour* stage accumulates information about the behaviour of features over a period of time. One way of doing this is to accumulate a four-dimensional histogram, the four dimensions of the histogram being x-position, y-position, x-velocity, y-velocity.

The *track classification* stage classifies each track as being normal or abnormal. One way of classifying a track is to compare the frequency of occupancy of the corresponding histogram cell with a threshold. If the frequency of occupancy is below the threshold, the track is classified as abnormal, otherwise it is considered normal.

The *alarm generation* stage generates an alarm signal when abnormal tracks are found to be present, subject to additional processing logic to resolve situations such as intermittent abnormal behaviour or multiple instances of abnormal behaviour associated with one real-world event, and other such situations.

Novel elements of the video motion anomaly detector are
1) The use of point feature extraction and tracking in an event detector
2) The detection of events by classification of feature behaviour as being abnormal, compared with the behaviour of features observed over time.

Compared with event detection based on normal video motion detection (so called), the video motion anomaly detector has the following advantages.
1) It is insensitive to changes in scene illumination levels, which are major source of false alarms in current video motion detectors (because it is based on point feature extraction rather detecting changes in image brightness).
2) It can detect the movement of small objects and raise an alarm if the movement is unusual (because it is based of point features rather than block processing).
3) It can detect movements of interest, even in the presence of other objects moving normally (because it accumulates information about feature behaviour).
4) It can be applied to a very wide range of different applications with little special setting-up (because it detects abnormal behaviour rather than pre-defined specific behaviour).

Compared with other existing event detections systems based on complex software solutions, the video motion anomaly detector is a simple system suitable for being implemented in inexpensive hardware.

Detailed description of the Invention

... TBC

I would appreciate some feedback on level of detail required etc and other material before drafting this section. – RJE.

Video Motion Anomaly Detector – Abstract

The Video Motion Anomaly Detector addresses the problem of automatically detecting events of interest to operators of CCTV systems used in security, transport and other applications, by processing CCTV images. The detector may be used, for example, to raise an alarm and summon a human operator to view video data, to trigger selective recording of video data or to insert an index mark in recordings of video data. The video motion anomaly detector extracts and tracks point-like features in video images and raises an alarm when a feature (or features) is (or are) behaving abnormally, compared with the behaviour of features observed over a period of time. Compared with existing event detectors called "video motion detectors" (devices which are essentially based on detecting changes in image brightness averaged over image sub-blocks), the video motion anomaly detector has the advantage of being less prone to false alarms caused by changes in scene illumination levels. The video motion anomaly detector can also detect the movement of smaller objects and detect movements of interest in the presence of other moving objects. Further, it can be applied to a very wide range of different applications with little special setting. Compared with other existing event detections systems based on complex software solutions, the video motion anomaly detector can be implemented inexpensive hardware.

Richard Evans
[redacted]